# 2 CAMERA-BASED FALL DETECTION SYSTEM WITH THE SERVICE ROBOT SANBOT ELF

J. Bauer [a, *], L. Gründel [a], J. Seßner [a], M. Meiners [a], M. Lieret [a], T. Lechler [a], C. Konrad [a], J. Franke [a]

[a] Institute for Factory Automation and Production Systems, Technical Faculty, Friedrich-Alexander University Erlangen-Nuremberg, Egerlandstr. 7-9, D-91058 Erlangen (Germany), jochen.bauer@faps.fau.de

**KEY WORDS:** Convolution Neural Network, Fall Detection, Keras, Machine Learning, Sanbot, TensorFlow

**ABSTRACT:**
Due to the demographic change and an aging population the care of the elderly is getting more and more attention, especially with the simultaneously increasing shortage of skilled workers in the health care sector. Improvements in the field of robotics are intended to help with these challenges. The use of service robots promises the possibility to reduce the overall costs of the health care system and enables older people to stay longer in their own homes. Falls among elderly lead to severe consequences for these people. In the course of this work a Convolutional Neural Network is deployed on the service robot Sanbot Elf to detect fallen people.

## 2.1 Introduction

### 2.1.1 Service Robotics

The International Federation of Robotics defines a service robot as a robot that performs useful tasks for people or things that are not considered industrial automation (International Federation of Robotics, 2016). For this reason, they are also called "non-industrial robotics" (Decker et al., 2011). The execution of service robots is either semi-autonomous with a robot human interaction or fully autonomous without operational intervention by the user (International Federation of Robotics, 2016). Humanoid service robots offer a further dimension of robot human interaction and are therefore expanding in many fields of application. Due to their human like body, they are better adapted to the human environment, thus leading to a higher acceptance among users (Haun, 2013). A robot is only able to develop human like traits or ultimately thought structures in this way, since the human being himself experiences and learns through the connection of his body to his environment (Haun, 2013). However, the acceptance does not increase with increasing human traits unceasingly. If a robot is adapted to the exterior of a human being, the acceptance of the user decreases again after a certain degree of similarity, which Masahiro Mori calls the *Uncanny Valley* (Mori et al., 2012).

The robot Sanbot Elf of the Chinese company QIHAN and the robot Pepper of the Japanese company SoftBank each represent a prototype of a humanoid service robot. The application of Pepper extends, among other things due to the high acquisition costs of approx. 20,000 euros (Generation Robots, 2018), primarily to the use in shops to welcome, inform and entertain customers (SoftBank Robotics, 2018). Sanbot Elf on the other hand enables further fields of application in the private sector due to its lower purchase price of about 10,000 euros (Cyber Robotics Technology Limited, 2018).

### 2.1.2 Challenges in Service Robotics

New technologies affect different social and societal levels. On a small scale, they influence the relationships between people or on a large scale social perspectives and traditions. In addition, the economic labor market situation also plays a role, as does the possibility of intervening in the development and introduction of these technologies in order to represent one's own interests. Becker et al. name here the impact on jobs, general technological development, acceptability and interpersonal communication as factors of social and societal challenges (Becker et al., 2013).

[*] Corresponding author.

D. Uckelmann (Ed.): Proceedings of the 1st Conference on Smart Public Buildings, pp. 15-28,
19-20 October 2018, University of Applied Sciences Stuttgart, Germany.
ISBN: 978-3-940670-66-3

#### 2.1.2.1 Impact on Jobs and Acceptability

In order to advance the dissemination of service robots, the greatest possible acceptance within the user groups is necessary. In this context, the professional user group in particular comes to the fore, whose acceptance is of decisive importance for the further establishment of service robots. In the health care sector, the care itself plays a significant role in this respect. Robots are accepted if there is a benefit for one's own work that makes it more efficient and pleasant. At the same time, however, a progressive loss of jobs due to the increasing use of robots is feared. Social upheavals are often driven by comprehensive technological changes. They influence human communication and behavior. The human being adapts to the new circumstances and is therefore rather a passive participant in this process (Becker et al., 2013).

#### 2.1.2.2 Human Communication

Due to technical achievements in recent decades, direct human communication is increasingly being supplemented and replaced by electronic transmission technologies. The widespread use of smartphones and social networks makes the exchange and connection between people uncomplicated and thus shifts communication to another level. In cities, there is a trend towards single and two-person apartments. As a result, older people are no longer necessarily supported by their direct relatives in the case of long-term care. Thus, direct contact with caregivers or assistants, in addition to their actual activities, is of great social importance (Melson, 2010).

In this context, an increasing use of service robots may contribute to two opposing effects. On the one hand, by relieving and assisting with monotonous and heavy work, caregivers may be able to engage more in personal interaction with those in need and thus play a significant social role. On the other hand, the possibility in the increased use of service robots may reduce direct interpersonal contact, as caregivers are less likely to be on site.

#### 2.1.2.3 Ethical and Legal Issues

When considering ethical aspects in the challenges of service robotics, two different fields have to be considered. The circumstances and conditions in the respective application case and the robotics itself. In the first case, the question arises as to whether the use of robots in the application area violates fundamental ethical values. If this is the case, the progressive spread and establishment of these robot systems will at least be made more difficult. The example of the health care system raises questions such as "dehumanization" in the care of elderly people by robots, as this is also referred to as inhuman. In addition, the impression may possibly arise that affected people are excluded from society or that they are regarded as test subjects (Becker et al., 2013). Such challenges need to be considered.

Ethical aspects concerning the robot itself raise the question of whether and to what extent ethical actions should be brought closer to them (Becker et al., 2013). Universally applicable principles do not exist, since they would be neither practicable nor feasible (Guo & Zhang, 2009). Robots are usually designed for a specific purpose, which is why it is probably sufficient to give them a code of ethics adapted to this area (Becker et al., 2013).

Ethical aspects frequently give rise to legal questions and should therefore always be seen in context with one another (Becker et al., 2013). In 1950 the science fiction author Isaac Asimov formulated the Four Laws of Robotics. However, conceived as universal principles, these laws reach their limits in some situations. Therefore, it is necessary to invest more effort and resources into changes of systems or the whole society in order to prevent the occurrence of ethical dilemmas (Helbing 2013, Helbing & Pournaras 2015).

### 2.1.3 Motivation

According to Carone & Costello (2006), the old-age dependency ratio in the European Union is expected to rise to 54% by 2050. This quotient describes the ratio between the number of people in the population group of 65-year-olds and older relative to the number of 15 to 64-year-olds. Thus, an ever larger group of elderly people is faced with an ever smaller group of young people. Against this background, the focus is increasingly on how to take care of these people. In this context, possibilities for longer care in one's own home also play a significant role. An important aspect here is the recognition of a fallen person. According to Halter et al. (2009), 87% of all fractures of older people are due to falls. According to statistics from the National Council on Aging in the USA, falls resulting in death also increase exponentially with age. This connection is shown in Figure 1. In the 65 to 69 age group, the fall rate is 5.4 (women) or 10.6 (men) per 100,000 inhabitants. In the following two groups, this rate rises to 19 and 34 respectively. However, people aged 85 and older are most severely affected. In this age group the fall rates show the highest values with 106.4 and 153.2 respectively. The difference in the values between women and men can be attributed to the fact that men are more likely to have comorbid conditions than women of the same age (Stevens, 2007). This presence of several diseases makes them more susceptible to the consequences of a fall (Stevens, 2007).

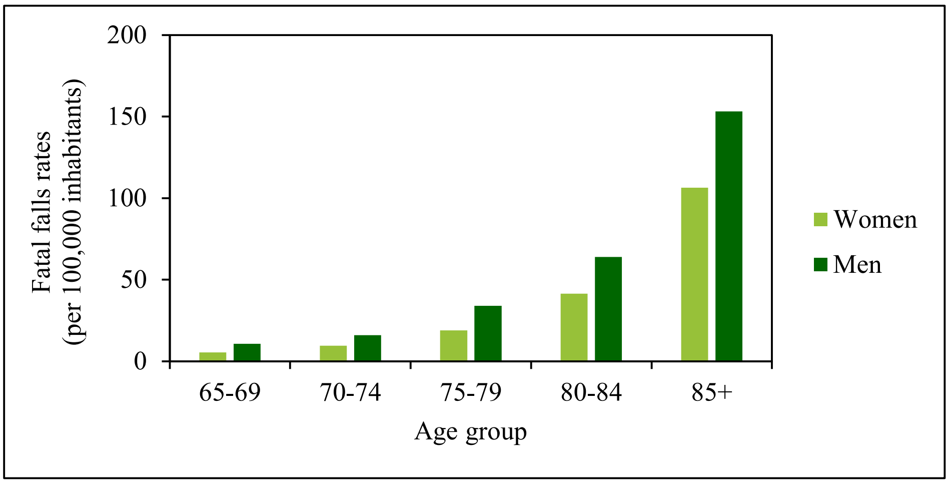

Figure 1. Fatal falls rate per 100,000 inhabitants in the USA in 2001, broken down into age groups and genders (Stevens, 2005)

It may be possible to mitigate the consequences of a fall and reduce the mortality rate by using a system that identifies a fallen person. The amount of time people lie on the ground after a fall is a significant factor influencing the severity of the effects of a fall (Igual et al., 2013). Many elderly people are no longer able to stand up on their own after a fall (Igual et al., 2013). Longer lying leads to symptoms such as painful pressure points, hypothermia or dehydration (Igual et al., 2013). A fall detection system tries to minimize the time spent lying on the ground by automatically initiating appropriate help measures after a fall has been detected.

Another positive effect of using such a system is the general safety it provides to people. After a fall, it is possible for people to develop a fear of falling again (Friedman et al., 2002). This fear of falling increases the risk of falling again (Friedman et al., 2002). It has possible negative effects such as depression, less activities and a generally lower quality of life (Scheffer et al., 2008). Therefore, a fall detection system offers possibilities to mitigate these effects and even prevent falls. In particular, the immediate environment of the elderly, such as either their own home or a care/retirement facility, plays an important role and is therefore defined in this paper as an area of application.

### 2.1.4 Robot Basics and Platforms

An industrial robot consists of a large number of components that are necessary for its operation. The manipulator represents the basic structure to which all other components are connected (Jazar, 2010). On the hardware side, actuators enable the movements of the robot, which change the configurations of the partial elements against internal and external forces in response to signals from the controller. The sensors of a robot are used to acquire information about the internal and external status.

The core processing unit serves as the core for monitoring and controlling the robot, since all information collected by the sensors converges at this point. Using the data, the processor plans the robot’s movements based on the kinematic limitations of the robot and determines how much and how fast the joints and limbs must move in order to achieve a desired position and speed. In addition, the processor monitors the activities carried out together by the controller and sensors (Niku, 2011).

Table 1 gives an overview of currently commercially available humanoid service robots as well as those that are still in the development stage or have been designed entirely for research purposes. Due to the rapid development in this field, it is not possible to present all systems comprehensively, which is why the table does not claim to be complete. Only exemplary current service robots for different areas are listed.

| | Name | Manufacturer | Main application area |
|---|---|---|---|
| Commercially available | BUDDY | Blue Frog Robotics | Companion |
| | Cobalt robot | Cobalt Robotics | Security |
| | OSHbot | Lowe's Innovation Lab | Retail |
| | Nao | SoftBank Robotics | Education, Research |
| | Pepper | SoftBank Robotics | Public |
| | wGO | Follow Inspiration S.A. | Retail, Public |
| | REEM | PAL Robotics | Public, Security |
| | Ramsee | Gama2Robotics | Security |
| | Sanbot | QIHAN | Public, Retail |
| Research robots | Care-O-bot | Fraunhofer IPA | Healthcare, Household |
| | Romeo | SoftBank Robotics | Healthcare, Household |
| | ASIMO | Honda | Research |
| | AMIGO | Eindhoven University of Technology | Healthcare, Household |
| | MOnarCH | Carlos III University of Madrid | Healthcare |
| | SPENCER | Albert-Ludwigs-Universität Freiburg | Airport |
| | Atlas, PETMAN | Boston Dynamics | Security |
| | Rollin' Justin | DLR | Household, Outer space |
| | Roboy | TU München | Public, Research |

Table 1. Overview of commercially available robots and research robots

### 2.1.5 Machine Learning Basics

With the continuously improved and more cost-effective hardware, more and more application fields and tasks are becoming possible. In order to enable (service) robots to perform complex tasks with a certain degree of autonomy, a form of independent learning is necessary. The field of machine learning offers suitable methods for this.

The development of an artificial neural network (ANN) resulted from the motivation to copy the information processing capabilities of a mammal's brain (Rojas, 1996). Based on the basic structure of neurons, ANNs consist of a multitude of artificial neurons or nodes (Haykin, 1999).

Mathematically, such a mesh is defined as a directed graph having the following characteristics (Bishop 2006, Müller et al. 1995):

- Input signals $x_i$ that are either the output signals of a node $i$ or initial data.
- A weight $w_{ij}$ for each connection between two nodes $i$ and $j$.
- One bias $b_j$ for each node $j$.
- An activation function $h$.

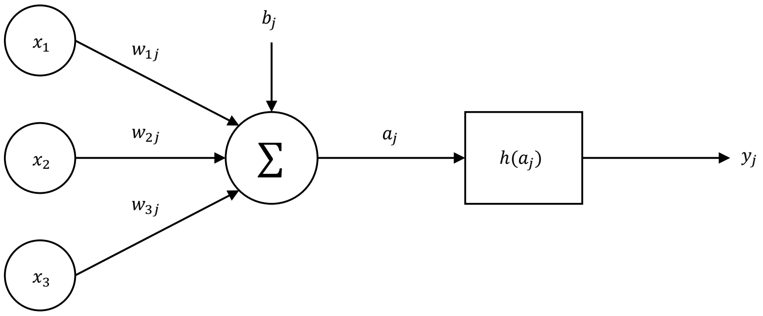

Figure 2. Schematic structure of a nonlinear artificial neuron (Haykin, 1999)

Figure 2 shows an example of a neuron $j$ where three input signals $x_1$, $x_2$ and $x_3$ are multiplied by a weight $w_{1j}$, $w_{2j}$ and $w_{3j}$ respectively. In addition, for each node a bias $b_j$ is added to the sum of these values. The subsequent intermediate result $a_j$ is converted into the output signal $y_j$ with an activation function $h(a_j)$. This activation function allows the creation of a nonlinear model.

Figure 3 shows the structure of a simple deep feedforward network that has a single hidden layer and is called *fully connected* because all nodes of a layer are connected to all nodes of the next layer. *Partially connected*, conversely, means that some of these connections are not present (Haykin, 1999).

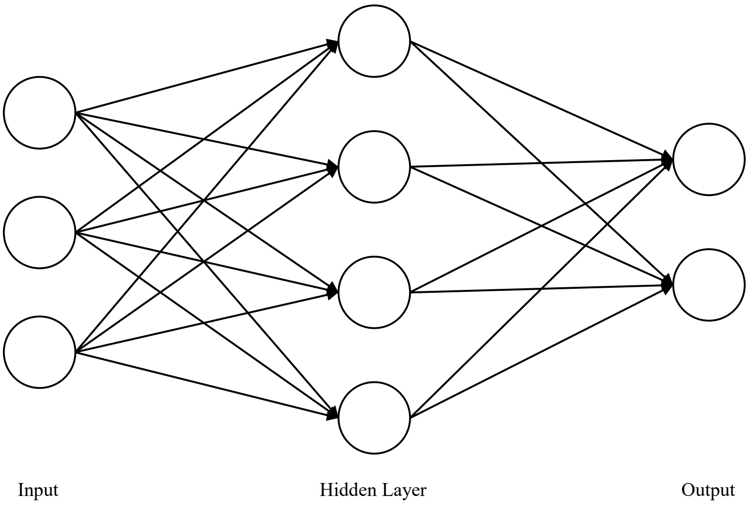

Figure 3. Construction of a simple 3-layer network with one hidden layer (Bishop, 2006)

The term *feedforward* has its origin in a peculiarity of the network. The information always flows in one direction from the input signals through the activation function to the result. Thus, there are no connections from a neuron which lead back to the neuron (Goodfellow et al., 2016).

Convolutional Neural Networks (CNN) originate from the work of Le Cun et al. (1989, 1998) and are mainly used to process data with a grid-like structure. Such a net differs from normal ANN by using a mathematical operation called convolution instead of a standard matrix multiplication in at least one layer (Goodfellow et al., 2016). Figure 4 shows an example of such a neural network.

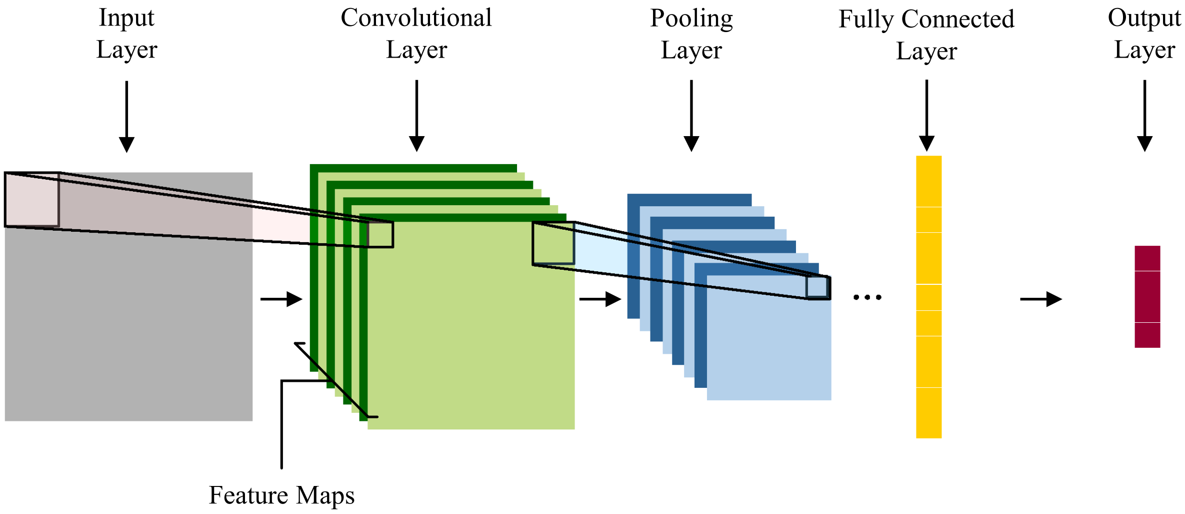

Figure 4. Typical structure of a CNN. An input layer is followed by convolutional and pooling layers, which end in a fully connected and output layer (Peng et al., 2016)

The goal is to create a model for the approximation of functions by statistical generalizations (Goodfellow et al., 2016). This model receives a training data set $S$ with an unknown distribution $D$, which is available at the beginning, in order to output a prediction $y$ (Shalev-Shwartz & Ben-David, 2016). The learning algorithm used in the model aims to minimize the error between its prediction and the actual data (Shalev-Shwartz & Ben-David, 2016). Training is done to improve the predictive accuracy of a model by creating an approximation function that is adjusted by minimizing a cost function.

## 2.2 Related Challenges

Due to the demographic change and the accompanying aging population, some areas, such as health care, are facing new challenges (Rothgang et al., 2016). The progressive robotization is intended to remedy this situation. Certain activities, previously typically performed by humans, are now delegated to service robots.

Developments in service robotics are currently at an intermediate stage. An ever greater acceptance of robots in the population contributes to higher volumes in the production of service robots, as a result of which prices also fall (Haun, 2013). This in turn leads to ever new potential areas of application in which in the past, due to the still high prices, the economic use of service robots was only possible to a limited extent. Nevertheless, only a few service robots are commercially available and affordable for the end user.

The majority of systems that are able to identify fallen persons are basically divided into two groups: Context-sensitive systems and portable devices. Context-sensitive systems use data provided by sensors, which are usually installed in a stationary manner in the desired environment. These include cameras, microphones, infrared and pressure sensors, and sensors in the ground. Portable devices are worn by the user directly or in close proximity to the body and detect a fall by means of the implemented sensors. Accelerometers are most frequently used for this purpose, which may also be supported by gyroscope sensors to determine the exact position of the fallen person (Igual et al., 2013).

Context-sensitive systems have the disadvantage that the sensors are stationary in the monitored environment. This means that unrestricted and flexible monitoring of the desired areas is not possible (Rougier et al., 2011). Portable devices, on the other hand, offer flexible and mobile monitoring because the sensors are mounted on or in the immediate vicinity of the body of the person being observed. However, the disadvantage is the obligation to carry the devices uninterruptedly in order to ensure continuous security through the system (Solbach & Tsotsos, 2017).

The Sanbot Elf combines the advantages of both approaches and avoids the respective disadvantages. Since the sensors are installed in the Sanbot, it is no longer necessary for them to be on or near the user's body. Furthermore, the Sanbot is able to move in the desired operating area, which is why the disadvantage of stationary sensors is not given. For these reasons, the Sanbot Elf is ideally suited to implement a system for detecting fallen persons using its cameras.

The use of ML methods results in a new approach to fall detection and better results than other classification tools (Yu et al., 2017). Especially the already mentioned CNNs are suitable for the processing and classification of image material. Using this type of ANNs on the Sanbot Elf promises a fall detection that does not have the disadvantages of context-sensitive and portable systems mentioned above.

## 2.3 Solution Concept

### 2.3.1 Robot Platform Sanbot Elf

The Sanbot Elf consists of 51 sensors that enable it to interact with its environment. A selection of these sensors is shown in Figure 5. The head unites most of the sensors that make these interactions possible. An HD color camera records videos and images and thus enables the perception of the surroundings. In addition to the color camera, a 3D camera is also embedded in the head of the Sanbot, which is able to capture depth information of the environment. The face is visualized by a low-resolution LED display on which the eye areas are simplified. LEDs are also mounted on the head, arms and underside, which glow in different colors. A total of seven microphones are mounted around the robot for participation in conversations, enabling a 360° localization of voices or noises. Another feature of the head is the rear mounted projector and an additional lamp on the front of the head that provides the Sanbot Elf with the ability to operate in dark environments. Furthermore, a total of seven tactile sensors are implemented on the front and top of the head. These sensors enable interaction with the service robot not only through voice commands, but also through specific touches. The entire head can be moved horizontally by 180° and vertically by 30° and is thus able to turn to localized voices (QIHAN Technology, 2018a, b).

The chest of the 19 kg Sanbot Elf is covered by a 10.1 inch Full-HD touch display, which serves as the main input device. There is another color camera below the screen, but with a lower resolution than the primary camera. At the bottom of the torso there are several touch sensors that are supplemented by infrared sensors. Thus, the Sanbot Elf is able to measure distances between itself and other objects and to use them for its movements. On the front are three loudspeakers, which enable the Sanbot to react to external influences and to communicate with people. Furthermore, inside the Sanbot Elf there is a gyroscope sensor as well as an electronic compass, which allow the determination of the Sanbot's orientation (QIHAN Technology, 2018a, b).

Additionally, there is a PIR sensor on the central part of the torso. Such a passive infrared sensor measures temperature fluctuations and is therefore capable of detecting movements (Song et al., 2008). For this reason, this sensor is used to detect and track people in front of the robot.

The service robot has an omnidirectional drive that enables it to move from any position in any direction. Moreover, this drive enables a 360° rotation on the spot and a maximum speed of 0.8 m/s. For the safe recognition and avoidance of obstacles, being persons or objects, infrared sensors installed at the bottom of the torso serve as bumpers. This ensures reliable movements in its surroundings, which do not result in any damage (QIHAN Technology, 2018a, b).

The operating system of the Sanbot Elf is Android 6 which is accessible through the integrated touch screen mentioned before. Due to this fact, the detection system for fallen persons is deployed as an Android app on the Sanbot Elf.

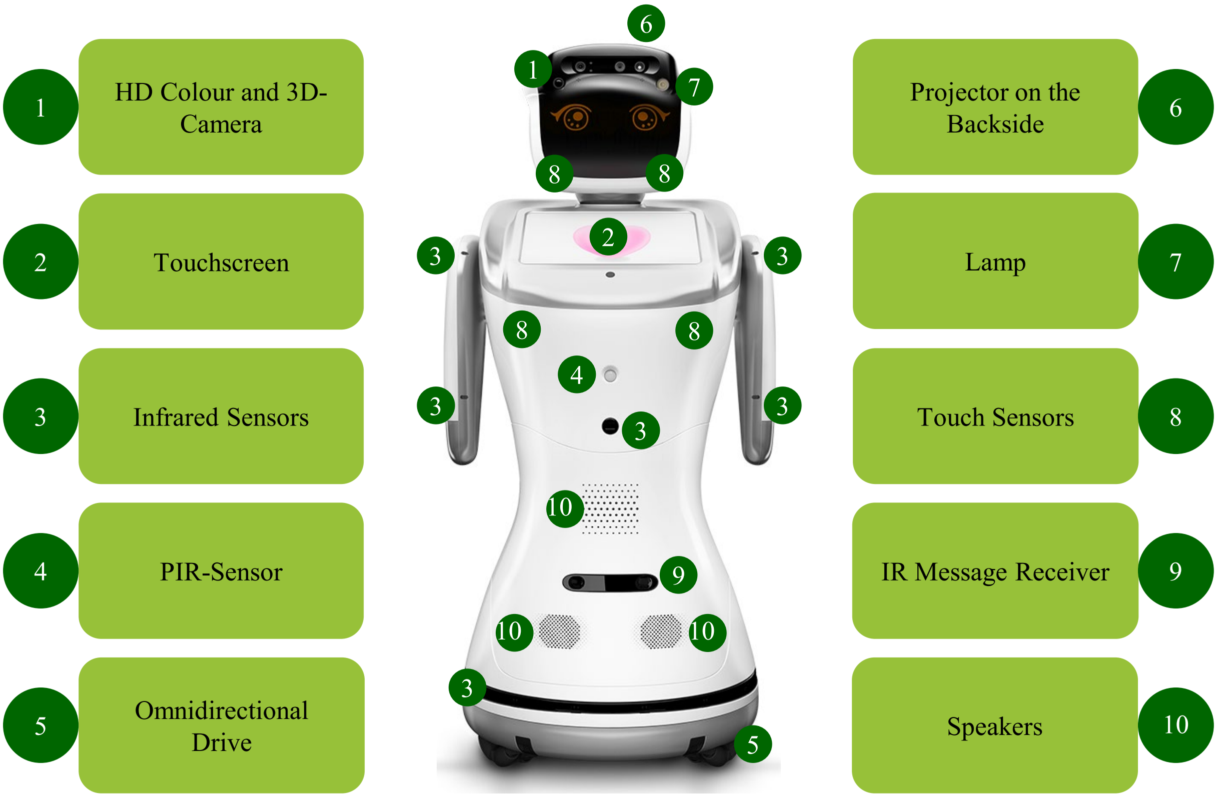

Figure 5. Hardware built into the Sanbot Elf. Picture of the Sanbot Elf: QIHAN Technology (2018a)

### 2.3.2 Process of the Fall Detection System

Figure 6 shows the basic concept of a system for the detection of fallen persons, which is developed in the course of this paper. Using the color camera integrated in the Sanbot Elf, the camera stream of the area to be monitored is recorded. The individual image data of this video stream is continuously processed and forwarded to a previously trained CNN. This CNN analyses the individual images and divides them into two classes:

- Class 0: *fallen*
- Class 1: *not fallen*

This classification takes place without interruption as long as the video stream of the camera is active. A fallen person, who is no longer able to stand up on his own, will remain on the ground for a long time. For this reason, a person will be considered to be in need of help if they remain on the ground for a predefined period of time. If the CNN classifies a person, who has fallen during this period on the basis of the image data, follow-up measures are initiated. The time span here is set to 60 seconds, as it should normally be possible for a fallen person to at least sit up during this time to no longer be classified as having fallen.

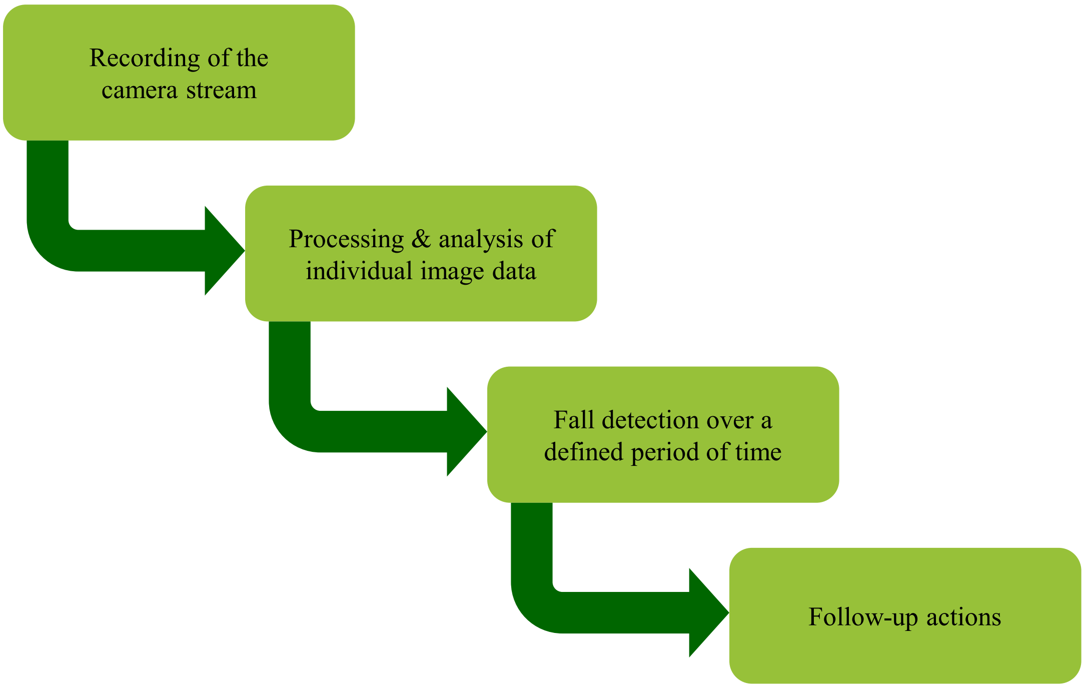
Recording of the camera stream
Processing & analysis of individual image data
Fall detection over a defined period of time
Follow-up actions

Figure 6. Approach for a solution system to detect fallen persons based on the analysis of image data obtained from a camera

## 2.4 Implementation

In addition to the IDE Android Studio 3.0.1 and the *QIHAN SDK 1.1.8* for app development, software for the development of neural networks is used. Therefore, PyCharm with the programming language Python is selected as the development environment. The creation and training of the neural networks is realized by means of the open source library Keras, which contains different ML libraries including the TensorFlow (TF) library. The training of the ANN then takes place on a mobile GPU. The implementation process is structured as follows:

1. Acquisition and preparation of training data
2. CNN model constellation
3. Training and optimization of the CNN
4. Porting the CNN to Android
5. Integrating the CNN into the app

As the name implies, in TF the ML algorithms are constructed as data flow graphs, where the nodes represent the operations performed and their connections (or edges) represent the data flow. Tensors (n-dimensional fields) are used between the nodes as a general exchange format containing primitive data types such as *int32*, *float32* and *string*. TF allows the user to realize applications on different platforms like distributed clusters, local workstations or mobile devices. The scripting language Python acts as a wrapper for the creation of graphs and allows the user to customize and extend the graphs and models without having to change the basic system (Abadi et al., 2016).

### 2.4.1 Acquisition and Preparation of Training Data

In principle, the information that is required at the beginning is the information that is predicted by the ANN. In the present case of the recognition of fallen persons, this is image data that depicts the state of a fallen person. As already described in section 2.3.2, CNN image data is classified into two classes: On the one hand, persons who have belong to class 0 (fall) and on the other hand persons who have not fallen (class 1). For this reason, image data for binary classification is necessary for both the positive and the negative class. No freely accessible data set with pictures of fallen persons that is suitable for this work is publically available, which is why a separate data set is created in the course of this work. This data set consists of images that are obtained via both a Google image search and by own images of people lying on the ground.

### 2.4.2 Network Structure

There are two different procedures for choosing a suitable CNN. One option is to either create your own model or use an existing model of a CNN and train it completely on the basis of the existing data set. This gives the user the greatest possible freedom in network structure. Complex models, however, sometimes reach millions of parameters, for whose training correspondingly large amounts of image data are required (Google LLC, 2018). This in turn also requires adequate computing power to carry out the training in an acceptable amount of time (Google LLC, 2018).

If one or both of these requirements are not met, another option is to use pretrained CNNs. These complex CNNs are already trained on large data sets with hundreds or thousands of different classes and can be retrained to new classes without retraining the weights of the entire model.

The CNN used in this paper to classify images is shown in Figure 7 and is based on the CNN image classification architectures proposed by Yann LeCun (LeCun & Bengio, 1995).

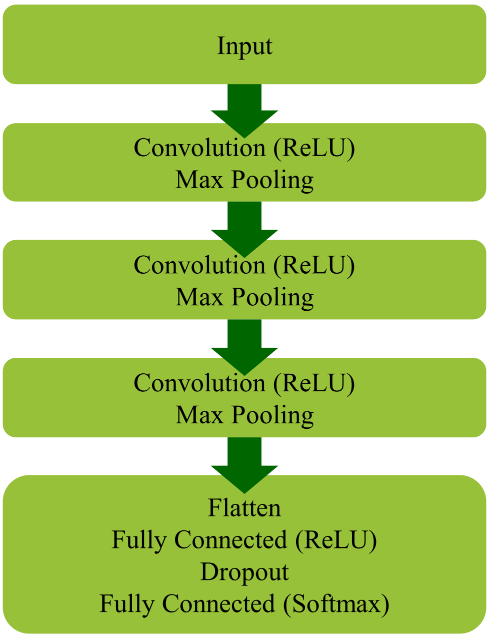

Figure 7. Construction of a CNN consisting of repeating blocks with convolutional and max pooling layers. An output block consisting of a flatten layer and a fully connected layer follows while dropout is used as regulation

A data set widely used in visual object recognition is provided by the ImageNet project. It contains 21,841 different classes consisting of a total of 14,197,122 images with corresponding labels (as of 30.04.2010) (Stanford Vision Lab, 2016). A CNN trained on the basis of this data set has thus learned helpful features for a wide range of classification problems. Therefore, it can be used for the classification of unknown objects as well. This circumstance is exploited by using such a CNN and retraining it to own classes by using a much smaller own data set. For this purpose, the last block for outputting the results is separated from the rest of the model and replaced by the same output block in Figure 7. This block contains the necessary configurations for the classification of the fallen persons.

With a total number of 1,064 images for class 0 only a limited data set is available. This is due to the effort involved in generating such image data. Consequently, the classification model in this paper is based on the use of an already trained CNN. By retraining this CNN, useful results can be achieved with the present limited data set.

The best-known CNNs, some of which are listed on the official TF website, that have achieved good results in past classification competitions are the *Inception, Inception-ResNet, NASNet, VGG16* and *ResNet* models. There are also *MobileNets* that have been developed especially for the use on devices that do not have large computing capacities (Howard et al., 2017). Furthermore, the use of the given energy on mobile device is a crucial factor because the higher the number of nodes in a model the higher the required energy for the classification. The focus here is on weighing accuracy against resource-saving performance. In this work, the *Inception v3* model (Figure 8) is retrained and used for the classification problem. This model is based on the work of Szegedy et al. (2016) and consists of blocks of layers arranged one after the other.

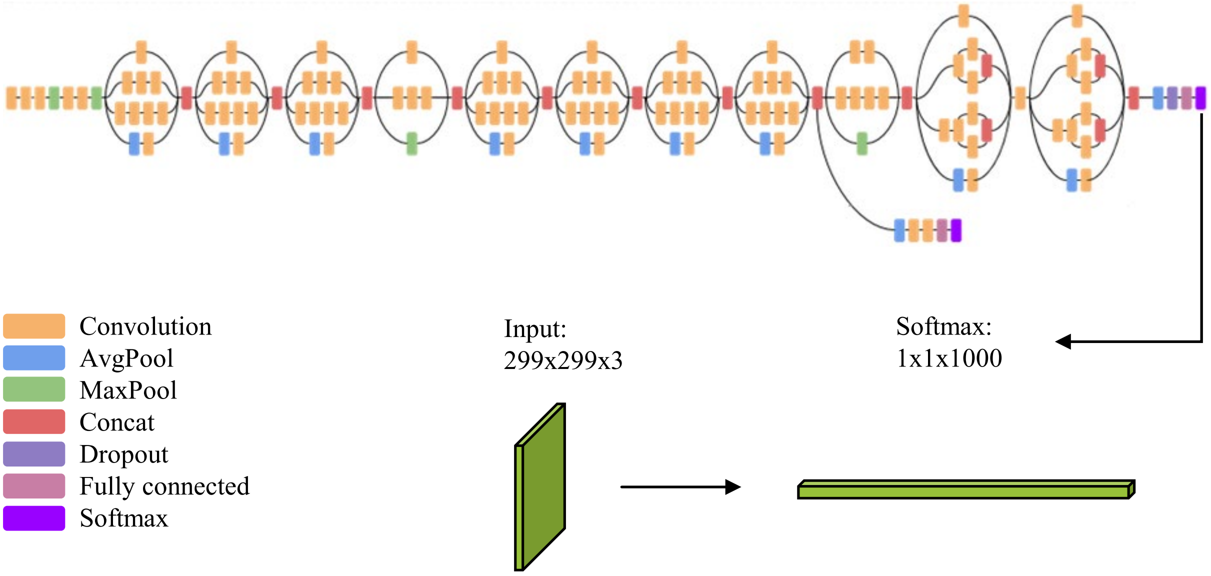

Figure 8. Structure of the Inception v3 model developed by Google. A layer of size 299x299x3 serves as input, followed by blocks of different layers to perform certain mathematical operations. The Softmax output layer of size 1x1x1000 is used to classify 1,000 different classes (Google Developers, 2018).

### 2.4.3 Training

At the beginning of each classification problem, the existing data set must be split into three separate subsets. With 688 images per class (66 percent), the training data set contains the majority of this data set. The remaining set is divided between the validation set with 320 images per class. The test data set contains 56 images for class 0 and 41 images for class 1. This is followed by the selection of a basic model whose weights have already been trained, for example using the *ImageNet* data set. Furthermore, the hyperparameters epochs and batch size as well as the input size of the images are specified. The existing image data is then preprocessed and multiplied by a data generator. The images are, among other things, rotated, scaled, distorted, enlarged or reduced. These random transformations of the images increase their total number and ultimately lead to better results.

The process of creating an overall model is shown in Figure 9. In order to be able to introduce own classes into the model, the top *Fully Connected Layers*, which are responsible for the actual classification, have to be separated from the rest of the model. For the classification, a superordinate model adapted to the problem is now created, trained and reunited with the base model. The previously generated image data is once routed through the basic model and processed. The resulting output is saved and then serves as an input layer for the higher-level model. Using these features, also known as bottleneck features, the actual training of the higher-level model is carried out. This leads to a more efficient training, since a run through the base model is bypassed in each iteration. Finally, the trained superordinate model is reunited with the base model. As a final step, the prediction accuracy is checked against the overall model using the test data set.

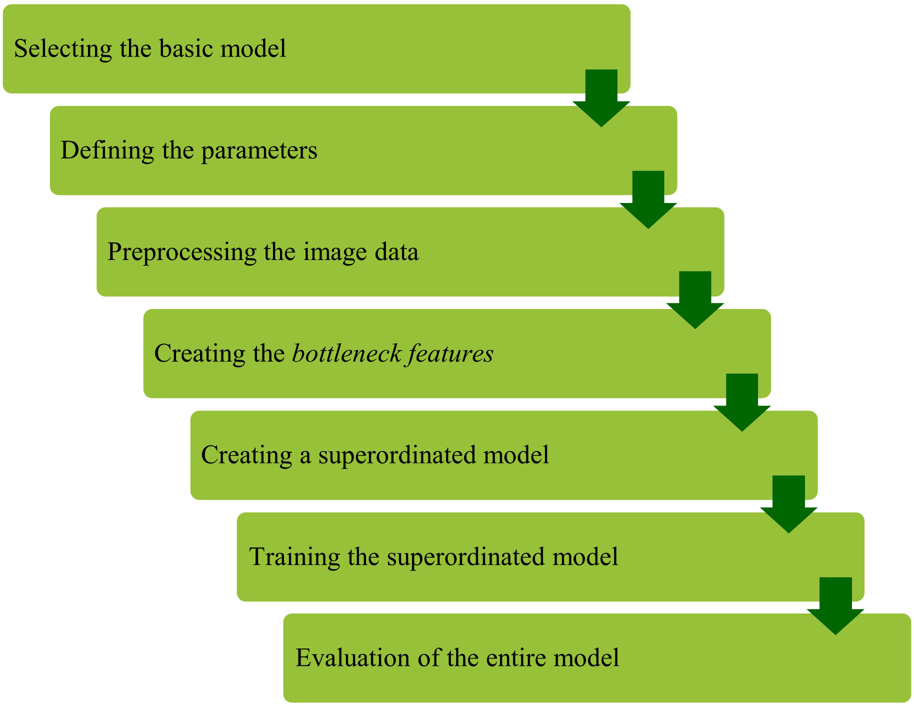
Selecting the basic model
Defining the parameters
Preprocessing the image data
Creating the *bottleneck features*
Creating a superordinated model
Training the superordinated model
Evaluation of the entire model

Figure 9. Process of creating an overall model for classifying your own image data using a previously trained basic model

### 2.4.4 Porting and Integrating the CNN into the App

Keras allows a simpler generation of neural network models. However, the resulting models are not suitable for immediate use with TF. In order to use this format again, it is necessary to convert the models from Keras' own format back to the TF format.

The application of a TF model to classify images on an Android or iOS operating system is possible with two different solutions: TF Mobile and TF Lite. In the course of this work TF Mobile is chosen due to the fact that the training is done on a Windows operating system and this solution offers more stability.

For the use of TF on Android devices there is a Google demo application (Google LLC, 2016) available, which includes four functions. One of these functions is TF Classify that makes simple classification of objects possible. The previously created files for the own CNNs and the corresponding labels are added to the project and now allow a classification based on the own defined classes. This function is then added to the app being deployed on the Sanbot Elf to detect fallen persons.

## 2.5 Results and Validation

### 2.5.1 Validation of the ANN

Due to existing difficulties in the app, the display of the camera stream and the classification are evaluated separately. For the classification a locally stored image of class 0 and 1 is used and analyzed by the algorithm and the CNN. This procedure shows the feasibility of the present fall detection and its use in the application, since both images are correctly classified.

This paper focuses on the application of an already trained model and its retraining to the desired classes. The CNN models *Inception v3* and *MobileNet v1* are being used. They are among the best-known and most frequently used models in past competitions. As a comparison, an additional model is used whose total weights are trained solely by the existing data set and is subsequently referred to as CNN1. The adjustment of these parameters to the respective conditions and their analysis is also referred to as *hyperparameter tuning*. A comprehensive tuning of the hyperparameters, however, is outside the focus of this paper, which is why the most common settings are used, which have proven to be promising in the past (Table 2).

| Hyperparameter | Value |
|---|---|
| Cost function | Categorical cross entropy |
| Epochs | 50 |
| Regulation | Dropout |
| Optimization | SGD (momentum = 0.9) |
| Learning rate | 0.0001 |
| Activation function | ReLU, Softmax |
| Batch size | 16 |
| Size of the Input Layer | 224x224x3 |

Table 2. Selection of the most important parameters for the CNN training

For the final evaluation of the models, the test data set is used. The results are shown in the confusion matrix in Table 3. The *Matthews Correlation Coefficient* (MCC) can be used to evaluate an ANN using the entire confusion matrix. A value of $MCC = 1$ describes a perfect prediction while $MCC = -1$ indicates a total contradiction between prediction and reality. A value of zero means that the result of the model is no better than random predictions (Matthews, 1975).

| Name | Accuracy | Costs | MCC |
|---|---|---|---|
| CNN1 | 0.92 | 0.24 | 0.80 |
| Inception v3 | 0.98 | 0.06 | 0.96 |
| MobileNet v1 | 0.99 | 0.06 | 0.94 |

Table 3. Results of the evaluation of the CNN1*, Inception v3* and *MobileNet v1*

The CNN1 is inferior to the *Inception v3* and *MobileNet v1* in all metrics, i.e. cost, accuracy and MCC. In comparison with the other models, the cost value is relatively high at 0.24 and the result of the test data set by the MCC comparatively low at 0.80. It should be noted that the model has only approximately 2.8 million weights, which are completely trained on the basis of a small data set. The *Inception v3* has 34.8 million weights whereas the *MobileNet v1* consists of 16 million weights.

The *Inception v3* and *MobileNet v1* both have a cost value of 0.06. This value can be attributed to the fact that in both models the base model has already been trained using 1,000 different classes of the ImageNet data set. As a result, the models have already learned helpful features for distinguishing objects that have a positive effect on the classification problem at hand. The final test for the *Inception v3* with a value of 0.96 for the MCC is slightly better than for the *MobileNet v1* with a value of 0.94. This can be explained, among other things, by the almost twice as high number of weights in the network. The performance of the newer *MobileNet v1* shows that the development of more powerful and more efficient ANN is increasing.

Based on the results obtained, the retrained *Inception v3* is used for implementation in the app, as it shows the best performance of all models. In the light of the limited battery capacity of the Sanbot Elf, it is also possible to use the *MobileNet v1*, which, with a minimally lower performance, enables a resource-saving deployment due to its reduced size. In a real application, however, it is advisable to use the *Inception v3*.

### 2.5.2 Validation on External Influences

In addition to persons who have fallen, the data set at hand represents a high variability of images deviating from this. In a domestic environment, however, it is likely that people will also adopt postures that may resemble those of a fallen person. For example, the difference between a sleeping person and a fallen person is not recognizable by the ANN. In addition, there are situations in which people sit on the floor or squat. In isolated cases items of clothing spread out on the ground, such as a bathrobe or a motorcycle suit, are classified as a fallen person by the CNN. This circumstance may be remedied by a data set that is increasingly oriented towards human poses. This also gives human body parts such as head, hands and feet a higher weight in the classification.

## 2.6 Discussion

The applications can be divided fundamentally into four areas, which are the subject of this chapter: retail, public, education as well as the health care sector, ambient assisted living (AAL) and smart home environments.

In shops, the Sanbot Elf plays the role of an information medium. Through its face recognition it is possible to locate customers entering a store and greet them directly. The customer is made aware of current products or offers, which are shown directly on the display through the integrated product preview.

The public area includes, apart from the retail mentioned above, freely accessible buildings such as public administration buildings, railway stations or airports. At airports and railway stations an application is conceivable to provide passengers with information about their travel times, ticket prices or current offers from the shops. Other areas such as police stations, courts or libraries are conceivable as general sources of information and aids (QIHAN Technology, 2018a).

The increasing digitalization does not stop at the education system and changes the classical teaching and its learning methods (Dräger & Müller-Eiselt, 2015). Due to its features, the Sanbot Elf is able to carry out a wide variety of activities. The built-in projector makes additional external hardware for presentations redundant, while the touch display of the Sanbot Elf makes it possible to interactively bring lesson content closer to the students.

The application of the Sanbot Elf in the health sector is one of the largest areas of application. Since health care extends to the private environment in addition to large facilities, the overlapping AAL and smart home environments are also included in this context. The Sanbot Elf offers the possibility of continuous monitoring and care of patients. In general, the service robot serves as a supporting assistance system for the workers as well as the people to be cared for. In senior citizens' and nursing homes, the Sanbot Elf serves not only as an assistant, such as supplying the residents with drinks, but is also used for entertainment by playing music or watching films via the built-in projector. In addition to facilities such as hospitals or retirement homes, the Sanbot Elf is also deployed in private homes. Thus, continuous care of people is also possible, without one person necessarily being on site all the time. This is especially helpful when affected persons are no longer able to provide for themselves independently. In addition, there is a reminder function that draws attention to take certain medications, the performance of required exercises or the adherence to doctor's appointments (QIHAN Technology, 2018a). The interaction possibilities of the Sanbot also make remote monitoring by doctors possible, who can then identify initial problems more quickly with the help of video chats.

## 2.7 References

Abadi, M., Barham, P., Chen, J., et al. (Eds.), 2016. TensorFlow: A System for Large-Scale Machine Learning (th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)).

Becker, H., Scheermesser, M., Früh, M., Treusch, Y., Auerbach, H., Hüppi, R. A., & Meier, F., 2013. Robotik in Betreuung und Gesundheitsversorgung. Vdf, Hochschulverlag AG an der ETH Zürich (TA-SWISS, 58), Zurich.

Carone, G., & Costello, D., 2006. Can Europe Afford to Grow Old? In: *Finance and Development* 43.

Cyber Robotics Technology Limited (Ed.), 2018. *Sanbot Elf.* https://www.robotics.com.hk/product/sanbot-elf/ (accessed May 26, 2018)

Decker, M., Dillmann, R., Dreier, T., Fischer, M., Gutmann, M., Ott, I., & Spiecker genannt Döhmann, I., 2011. Service robotics: do you know your new companion? Framing an interdisciplinary technology assessment. In: *Poiesis & Praxis* 8 (1), pp. 25–44. DOI: 10.1007/s10202-011-0098-6

Dräger, J., & Müller-Eiselt, R., 2015. Die digitale Bildungsrevolution: Der radikale Wandel des Lernens und wie wir ihn gestalten können. Deutsche Verlags-Anstalt (DVA), München.

Generation Robots (Ed.), 2018. *Pepper for Business Edition.* https://www.generationrobots.com/en/402422-pepper-for-business-edition-humanoid-robot-2-years-warranty.html (accessed May 26, 2018)

Google Developers, 2018. *Cloud TPU. Advanced Guide to Inception v3 on Cloud TPU.* https://cloud.google.com/tpu/docs/inception-v3-advanced (accessed July 11, 2018)

Google LLC, 2016. *TensorFlow Android Camera Demo. Edited by TensorFlow.* https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/android (accessed July 18, 2018)

Google LLC, 2018. *TensorFlow. Image Retraining.* https://www.tensorflow.org/hub/tutorials/image_retraining (accessed July 11, 2018)

Guo, S., & Zhang, G., 2009. Robot rights. In: *Science (New York, N.Y.)* 323 (5916), p. 876. DOI: 10.1126/science.323.5916.876a

Halter, J. B., Ouslander, J. G., Studenski, S., High, K. P., Asthana, S., Supiano, M. A., & Ritchie, C. S., 2009. Hazzard's geriatric medicine and gerontology. 6th ed. McGraw-Hill Education Medical, New York.

Haun, M., 2013. Handbuch Robotik. Programmieren und Einsatz intelligenter Roboter. 2., Aufl. 2013. Springer Berlin (VDI-Buch), Berlin.

Helbing, D., 2013. Globally networked risks and how to respond. In: *Nature* 497 (7447), p. 51.

Helbing, D., & Pournaras, E., 2015. Society: Build digital democracy. In: *Nature News* 527 (7576), p. 33.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al., 2017. Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications. In: *arXiv preprint arXiv:1704.04861*

International Federation of Robotics, 2016. *World Robotics. Service Robots. Chapter 1.2.* https://ifr.org/img/office/Service_Robots_2016_Chapter_1_2.pdf (accessed May 22, 2018)

LeCun, Y., & Bengio, Y., 1995. Convolutional Networks for Images, Speech, and Time-series. In: *The handbook of brain theory and neural networks* 3361 (10), p. 1995.

Matthews, B. W., 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. In: *Biochimica* et *Biophysica Acta (BBA) - Protein Structure* 405 (2), pp. 442–451. DOI: 10.1016/0005-2795(75)90109-9

Melson, G., 2010. Child development robots: Social forces, children’s perspectives 11, pp. 227–232.

Mori, M., MacDorman, K. F., & Kageki, N., 2012. The Uncanny Valley: The Original Essay by Masahiro Mori. In: IEEE *Robotics & Automation Magazine*, pp. 98–100.

QIHAN Technology, 2018a. http://en.sanbot.com/ (accessed June 6, 2018)

QIHAN Technology, 2018b. *Robot S1-B2 User Manual. V1.3.* http://en.sanbot.com/support/ (accessed June 14, 2018)

SoftBank Robotics (Ed.), 2018. *Who is Pepper?* https://www.softbankrobotics.com/emea/en/robots/pepper (accessed May 26, 2018)

Song, B., Choi, H., & Lee, H. S., 2008. Surveillance Tracking System Using Passive Infrared Motion Sensors in Wireless Sensor Network. In: *International Conference on Information Networking, 2008*. ICOIN 2008; 23–25 Jan. 2008, Busan, Korea (South). 2008 International Conference on Information Networking. Busan, South Korea, 1/23/2008–1/25/2008. International Conference on Information Networking; ICOIN. Piscataway, NJ: IEEE Service Center, pp. 1–5.

Stanford Vision Lab, 2016. *ImageNet database.* http://image-net.org/ (accessed July 11, 2018)

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (Eds.), 2016. Rethinking the Inception Architecture for Computer Vision. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 27–30 June 2016.